CLAIMS

What is claimed is:

1. A method comprising:

receiving information regarding an atomic distributed transaction, the atomic distributed transaction representing an aggregation of a plurality of discrete transactions for resource items that span a plurality of network resources;

placing a tentative hold on each of the plurality of resource items by causing a tentative hold record to be created and associated with each of the plurality of discrete transactions, the tentative holds operating in a non-mutually exclusive manner, thereby allowing the same resource item to be tentatively held by more than one transaction; and

after successfully gaining the tentative holds on each of the plurality of resource items and receiving a confirmation regarding the atomic distributed transaction, attempting to direct the completion of the atomic distributed transaction by conventional means.

2. The method of claim 1, wherein said attempting to direct the completion of the atomic distributed transaction by conventional means comprises initiating conventional Two-Phase Commit (2PC) prepare and commit processing for each of the plurality of discrete transactions.

3. The method of claim 1, further comprising receiving a notification indicating one of the plurality of discrete transactions are no longer possible.

4. The method of claim 1, wherein one or more of the tentative hold records are stored at an intermediate server that is not within the enterprise offering the resource item.

5. The method of claim 1, wherein the plurality of network resources comprise database systems of a plurality of different enterprises.

6. A method comprising:

receiving information regarding a distributed transaction from an originating application, the distributed transaction involving a plurality of items spanning a plurality of network resources; and

initiating a tentative-hold processing stage by requesting that a plurality of resource managers residing on one or more remote servers and participating in the distributed transaction each tentatively hold an item of the plurality of items involved in the distributed transaction and store call back information identifying a return communication path to the originating application, the tentative hold records operating in a non-mutually exclusive manner, thereby allowing items associated with the one or more remote servers to be tentatively held by more than one application.

7. The method of claim 6, wherein at least two of the remote servers are associated with different enterprises.

8. The method of claim 6, further comprising receiving a commitment corresponding to the distributed transaction from the originating application; and responsive to the commitment, initiating a two-phase commit processing stage by directing the resource managers to reserve the items during which the resource managers reserve the items and notifying, via corresponding call back information, other applications having a tentative hold on the same items that their respective tentative holds have been suspended.

9. A method comprising:

receiving, from a first client, a first request associated with a first discrete transaction, the first request soliciting a non-mutually exclusive hold on a resource item; the resource item being part of a first atomic distributed transaction that spans a plurality of network resources;

maintaining a first non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a first tentative hold record to be created and associated with the resource item and initiating a first timeout associated with the first tentative hold record;

receiving, from a second client, a second request associated with a second discrete transaction, the second request soliciting a non-mutually exclusive hold on the resource item, the resource item being part of a second atomic distributed transaction;

maintaining a second non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a second tentative hold record to be created and associated with the resource item and initiating a second timeout associated with the second tentative hold record;

receiving, from the first client, a third request associated with the first discrete transaction, the third request asking that completion of the first discrete transaction commence; and

responsive to the third request, suspending the second non-mutually exclusive hold and granting an exclusive lock on the resource item to the first discrete transaction.

10. The method of claim 9, wherein at least two network resources of the plurality of network resources are associated with different enterprises.

11. The method of claim 9, further comprising:

storing call back information associated with an application originating the second discrete transaction; and

notifying the application regarding the suspension of the second non-mutually exclusive hold.

12. The method of claim 9, further comprising in response to a timeout on the exclusive lock, recommencing the second non-mutually exclusive hold on behalf of the second discrete transaction.

13. A distributed transaction processing system comprising:

a distributed transaction coordinator executing on a first client system, the distributed transaction coordinator to place non-mutually exclusive holds on each of a plurality of resource items associated with an atomic distributed transaction that spans a plurality of network resources and to commence completion of the atomic distributed transaction by obtaining exclusive locks on each of the plurality of resource items after non-mutually exclusive holds have been successfully granted on each of the plurality of resource items; and

a distributed transaction manager executing on a server system communicatively coupled with a plurality of client systems including the first client system, the distributed transaction manager to maintain a plurality of non-mutually exclusive holds for each of a plurality of resource items associated with the server system and to grant only one exclusive lock per single resource item of the plurality of resource items at a given time in response to requests from distributed transaction coordinators.

14. The distributed transaction processing system of claim 13, wherein the distributed transaction coordinator includes a Two-Phase Commit transaction coordinator.

15. The distributed transaction processing system of claim 13, further comprising one or more Two-Phase Commit resource managers communicatively coupled with the distributed transaction manager.

16. A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to:

receive information regarding an atomic distributed transaction, the atomic distributed transaction representing an aggregation of a plurality of discrete transactions for individual resource items that span a plurality of network resources;

place a tentative hold on each of the plurality of individual resource items by causing a tentative hold record to be created and associated with each of the plurality of discrete transactions, the tentative holds operating in a non-

mutually exclusive manner, thereby allowing the same resource item to be tentatively held by more than one interested party; and

after successfully gaining the tentative holds on each of the plurality of individual resource items and receiving a confirmation regarding the atomic distributed transaction, attempt to direct the completion of the atomic distributed transaction by conventional means.

17. The machine-readable medium of claim 16, wherein said attempt to direct the completion of the atomic distributed transaction by conventional means comprises initiating conventional Two-Phase Commit (2PC) prepare and commit processing for each of the plurality of discrete transactions.

18. The machine-readable medium of claim 16, wherein one or more of the tentative hold records are stored at an intermediate server that is not within the enterprise offering the resource item.

19. The machine-readable medium of claim 16, wherein the plurality of network resources comprise database systems of a plurality of different enterprises.

20. A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to:

receive, from a first client, a first request associated with a first discrete transaction, the first request soliciting a non-mutually exclusive hold on a resource item; the resource item being part of a first atomic distributed transaction that spans a plurality of network resources;

maintain a first non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a first tentative hold record to be created and associated with the resource item and initiating a first timeout associated with the first tentative hold record;

receive, from a second client, a second request associated with a second discrete transaction, the second request soliciting a non-mutually exclusive hold on the resource item, the resource item being part of a second atomic distributed transaction;

maintain a second non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a second tentative hold record to be created and associated with the resource item and initiating a second timeout associated with the second tentative hold record;

receive, from the first client, a third request associated with the first discrete transaction, the third request asking that completion of the first discrete transaction commence; and

responsive to the third request, suspend the second non-mutually exclusive hold and grant an exclusive lock on the resource item to the first discrete transaction.

21. The machine-readable medium of claim 20, wherein at least two network resources of the plurality of network resources are associated with different enterprises.

22. The machine-readable medium of claim 20, wherein the sequences of instructions further include instructions which, when executed by the processor, cause the processor to:

store call back information associated with an application originating the second discrete transaction; and

notify the application regarding the suspension of the second non-mutually exclusive hold.

23. The method of claim 20, wherein the sequences of instructions further include instructions which, when executed by the processor, cause the processor to recommence the second non-mutually exclusive hold on behalf of the second discrete transaction in response to a timeout on the exclusive lock.